Chapter 4
Resources Developed in the Autonomata Projects

Henk van den Heuvel, Jean-Pierre Martens, Gerrit Bloothooft, Marijn Schraagen, Nanneke Konings, Kristof D'hanens, and Qian Yang

4.1 Introduction

In many modern applications such as directory assistance, name dialing, car navigation, etc. one needs a speech recognizer and/or a speech synthesizer. The former to recognize spoken user commands and the latter to pronounce information found in a database. Both components need phonemic transcriptions of the words to recognize/pronounce, and since many of these words are names, having good automatic phonemic transcription of names is crucial for application development.

A problem, especially in view of the recognition of names, is the existence of different pronunciations of the same name. These pronunciations often depend on the background (mother tongue) of the user. Typical examples are the pronunciation of foreign city names, foreign proper names, etc. The first goal of Autonomata was, therefore, to collect a large number of name pronunciations and to provide manually checked phonetic transcription of these name utterances. Together with meta-data for the speakers, such data is a valuable resource in the research towards a better name recognition.

In order to develop an application, the developer further needs a tool that accepts words/sentences and that returns the phonetic transcriptions of these words/sentences. The second goal of the Autonomata project was to develop a tool that incorporates a state-of-the-art grapheme-to-phoneme convertor (in our case

H. van den Heuvel (✉) · N. Konings
CLST, Radboud University, Nijmegen, The Netherlands
e-mail: H.vandenHeuvel@let.ru.nl

J.-P. Martens · K. D'hanens · Q. Yang
ELIS, Gent University, Gent, Belgium
e-mail: Jean-Pierre.Martens@elis.ugent.be

G. Bloothooft · M. Schraagen
UiL-OTS, Utrecht University, Utrecht, The Netherlands
e-mail: G.Bloothooft@uu.nl

P. Spyns and J. Odijk (eds.), *Essential Speech and Language Technology for Dutch*, Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-30910-6_4,

from Nuance), as well as a dedicated phoneme-to-phoneme (p2p) post-processor which can automatically correct some of the mistakes which are being made by the standard g2p. Dedicated p2p post-processors were developed for person names and geographical names.

In the follow-up Autonomata project (Autonomata TOO[1]) the aim was to build a demonstrator version of a Dutch/Flemish Points of Interest (POI) information providing business service, and to investigate new pronunciation modeling technologies that can help to bring the spoken name recognition component of such a service to the required level of accuracy. In order to test the technology for POIs a speech database designed for POI recordings was needed and compiled.

In this contribution we will describe in more detail the four resources briefly sketched above that were developed in Autonomata and in Autonomata Too[2]:

1. The Autonomata Spoken Names Corpus (ASNC)
2. The Autonomata transcription Toolbox
3. The Autonomata P2P converters
4. The Autonomata TOO Spoken POI Corpus

Another contribution in this book (Chap 14, p. 251) will address the research carried out in the Autonomata projects.

4.2 The Autonomata Spoken Names Corpus (ASNC)

4.2.1 Speakers

The ASNC[3] includes spoken utterances of 240 speakers living in the Netherlands (NL) or in Flanders (FL). The speakers were selected along the following dimensions:

1. Main region: 50 % persons living in the Netherlands and 50 % living in Flanders
2. Nativeness: 50 % native speakers of Dutch and 50 % non-native speakers
3. Dialect region of *native* speakers: four dialect regions per main region
4. Mother tongue of *non-native* speakers: three mother tongues per main region
5. Speaker age: one third younger than 18
6. Speaker gender: 50 % male, 50 % female

[1]Too stands for Transfer Of Output. Autonomata TOO used the output of the first project to demonstrate the potential of the technology.

[2]Partners in both projects were Radboud University Nijmegen (CLST), Gent University (ELIS), Utrecht University (UiL-OTS), Nuance, and TeleAtlas. Autonomata lasted from June 2005 to May 2007; Autonomata Too lasted from February 2008 to July 2010.

[3]Section 4.2 is largely based on [4].

We aimed to recruit non-native speakers that still speak their (foreign) mother tongue at home and that have a level A1, A2 or B1 (CEF standard[4]) for Dutch. However, the above strategy appeared to be too restrictive given the limited amount of time there was to finish the speaker recruitment. Another problem was that Flemish schools do not work with the CEF standard. Nevertheless, whenever the CEF information was available, it was recorded and included in the speaker information file.

The 60 non-native speakers in a region were divided into three equally large groups. But since French is obviously an important language in Flanders and far less important in the Netherlands, the division in subgroups has been made differently in the two main regions:

- In **Flanders,** speakers with an **English, French** and **Moroccan** (Arabic) mother tongue were selected.
- In the **Netherlands,** speakers with an **English, Turkish** and **Moroccan** (Arabic) mother tongue were selected.

As foreign speakers mostly live in the big cities and as the dialect region they live in is expected to have only a minor influence on their pronunciation, the dialect region was no selection criterion for these speakers. Native speakers on the other hand were divided in groups on the basis of the dialect region they belong to. A person is said to belong to a certain dialect region if s/he has lived in that region between the ages of 3 and 18 and if s/he has not moved out of that region more than 3 years before the time of the recording. We adopted the same regions that were also used for the creation of the CGN (Spoken Dutch) corpus.[5]

The speaker selection criteria altogether resulted in the speaker categorization shown in Table 4.1.

4.2.2 Recording Material

Each speaker was asked to read 181 proper names and 50 command and control words from a computer screen. The command words are the same for every speaker, but in each region, the names read by a speaker are retrieved from a long list of 1,810 names. These lists were created independently in each region, meaning that there is only a small overlap between the names in the two long lists. Once created, the long list was subdivided in ten mutually exclusive short lists, each containing 181 names: 70 % names that are typical for the region (NL/FL) and 30 % names that are typical for the mother tongues covered by the foreign speakers (10 % for each mother tongue). The typical names for a region were further subdivided in 50 % frequent and 50 % less frequent words.

[4] http://en.wikipedia.org/wiki/Common_European_Framework_of_Reference_for_Languages

[5] http://lands.let.ru.nl/cgn/doc_Dutch/topics/version_1.0/metadata/speakers.htm

Table 4.1 Speaker distribution in the spoken name corpus

Region	Origin	Dialect region
120 Dutch (50 % males)	60 natives	15 WestDutch
		15 Transitional region
		15 Northern
		15 Southern
	60 non-natives	20 English
		20 Turkish
		20 Moroccan
120 Flemish (50 % males)	60 natives	15 Antwerp and Brabant
		15 East Flanders
		15 West Flanders
		15 Limburg
	60 non-natives	20 English
		20 French
		20 Moroccan

For the native speakers we used all ten short lists, meaning that each name is pronounced by six native speakers of a region. For the non-native speakers we worked with only six short lists in order to achieve that the same name occurs three or four times in each non-native subgroup (right column of Table 4.1).

For all languages except Moroccan we selected 25 % person names (each person name consists of a first name and a family name), 35 % street names and 15 % town or city names. We selected more street names than city names because there are – logically – more streets than cities in a country. For the Moroccan names, we chose to select only person names because Dutch speakers will only rarely be confronted with Moroccan geographical names. Furthermore, we adopted the French way of writing for Moroccan names.

Exonyms were not included; meaning that we selected "Lille" instead of "Rijsel". Acronyms for highways (e.g. E40, A12) were not selected either.

We also took care that all different standard elements like street, drive, avenue... are present in a proportional way.

Since first names and family names naturally go together, it was decided to select a first name and a family name of the same language of origin and the same frequency class (in case of typical Dutch names).

Since it may be interesting to investigate whether speaker-specific pronunciation phenomena can be derived to some extent from a restricted set of adaptation data, it was decided to let every speaker also pronounce a list of 50 words that are often encountered in the context of an application and that reveal a sufficient degree of acoustic variability to make the word utterances also suitable for acoustic model adaptation. A list of 50 such words was delivered by Nuance (cf. Table 4.2). It consists of 15 digit sequences and 35 common command and control words.

Table 4.2 Commands and control words included in the ASNC

0 7 9 1	9 0 2 3	sluiten	opnemen	netwerk
3 9 9 4	9 5 6 0	bevestigen	programmeren	infrarood
0 2 8 9	0 1 2 3	controleren	microfoon	instellingen
5 6 9 4	1 6 8 3	help	stop	herhaal
2 3 1 4	7 8 2 6	ga naar	opslaan	opnieuw
7 8 9 0	activeren	aanschakelen	macro	menu
2 2 2 3	annuleren	Nederlands	controlemenu	opties
5 6 7 8	aanpassen	herstarten	status	lijst
9 0 7 4	ga verder	spelling	batterij	Vlaams
3 2 1 5	openen	cijfer	signaalsterkte	Frans

4.2.3 Recording Equipment and Procedure

The speakers were asked to pronounce an item that was displayed in a large font on a computer screen in front of them. Every participant had to read 181 name items (cf. Sect. 4.2.2) and 50 command word items. To simulate the fact that in a real application environment, the user usually has some idea of the name type s/he is going to enter, the participants in our recordings were also given background information about the origin of the names. To that end, the name items were presented per name category: Dutch person names, English person names, Dutch geographical names, etc. The name category was displayed before the first name of that category was prompted.

For the presentation and recording we used software that is commonly used by Nuance for the collection of comparable speech databases.

The microphone was a Shure Beta 54 WBH54 headset supercardoid electret condenser microphone. A compact four Desktop audio mixer from Soundcraft was used as a pre-amplifier. The 80 Hz high-pass filter of the audio mixer was inserted in the input path as a means for reducing low frequency background noise that might be present in the room.

The speech was digitized using an external sound card (VXPocket 440) that was plugged into a laptop. The digital recordings were immediately saved on hard disk. The samples were stored in 16 bit linear PCM form in a Microsoft Wave Format. The sample frequency was 22.05 kHz for all recordings. Before and after every signal there is supposed to be at least 0.5 s of silence (this instruction was not always followed rigorously).

In Flanders, a large part of the recordings were made **in studios** (especially those of non-native speakers and adult speakers), the rest was made **in schools** (those of young speakers and non-natives who take courses in a language center). Recordings in schools may be corrupted by background noise and reverberation. In the Netherlands all recordings were made on location, mostly in schools.

4.2.4 Annotations

Each name token has an orthographical and four broad phonemic transcriptions (cf. Sect. 4.1). Two transcriptions were automatically generated by the Dutch and Flemish versions of the Nuance g2p, respectively. A hand crafted example transcription that is supposed to represent a typical pronunciation of the name in the region of recording was created by a human expert. Finally, an auditory verified transcription was produced by a person with experience in making phonemic transcriptions of speech recordings. All phonemic transcriptions consist of phonemes, word boundaries, syllable boundaries and primary stress markers. The automatically generated transcriptions were converted from the Nuance internal format to the CGN format.[6]

Obviously, the first three transcriptions are the same for all utterances of the same name in one region, and as a consequence, they are provided in the name lists, together with the orthography and the type and language of origin of the name.

The auditory verified transcriptions are specific for each utterance. These transcription files were made in Praat.[7] The annotator could listen to an utterance as many times as s/he wished, and s/he was asked to modify (if necessary) the example transcription that was displayed above the signal. The modification was done according to rules outlined in a phonemic transcription protocol that is distributed together with the corpus.

For the sake of consistency we chose to work with example transcriptions for all names, even though for foreign names spoken by native Dutch/Flemish speakers and Dutch/Flemish names spoken by foreigners these standard transcriptions do not really offer a time gain compared to transcribing from scratch.

4.2.5 Corpus Distribution

The corpus is 9 GB large and is distributed by the Dutch HLT-agency (TST-centrale).[8] The corpus has a rich body of documentation. There is a general documentation file describing all aspects of the corpus construction as well as the format and content of all corpus files. The documentation also contains the phonemic transcription protocol (in Dutch) that was used for the creation of the example transcriptions and the auditory verified transcriptions, as well as a translation of that protocol in English, Also included is a document (in Dutch) describing the internal validation experiments that were carried out in the course of the corpus construction process.

[6] http://lands.let.ru.nl/cgn/doc_English/topics/version1.0/formats/text/fon.htm

[7] http://www.praat.org

[8] http://www.tst-centrale.org/nl/producten/corpora/autonomata-namencorpus/6-33

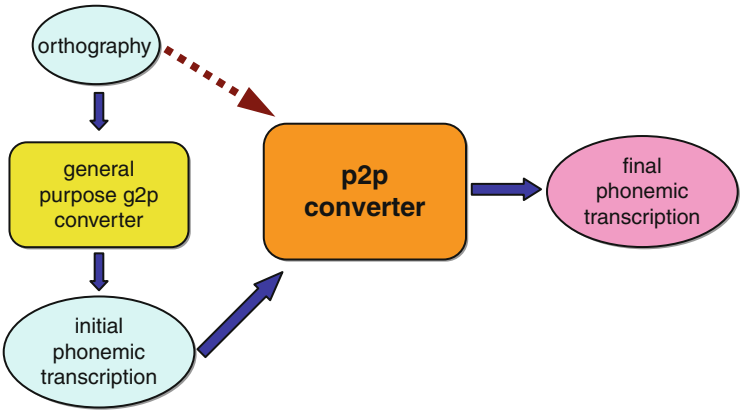

Fig. 4.1 Architecture of a two-step g2p converter

4.3 The Autonomata Transcription Toolbox

This toolset consists of a grapheme-to-phoneme (g2p) transcription tool and a phoneme-to-phoneme (p2p) learning tool.[9] The transcription tool is designed to enrich word lists with detailed phonetic transcriptions. It embeds the state-of-the-art general purpose g2p converters of Nuance (for northern and southern Dutch, English, French and German) and it can upload one or more specialized phoneme-to-phoneme (p2p) converters that were created by the p2p learning tool and that were designed to improve the outputs of the general-purpose g2p converter for names from a specific domain (e.g. street names, POIs, brand names, etc.). The p2p learning tool offers the lexicon developer the means of creating suitable p2p converters from a small lexical database of domain names and their correct transcription (see [4, 6, 7]). The p2p converters can be configured to generate multiple pronunciations with associated probabilities.

4.3.1 A Two-Step g2p Converter Strategy

The general architecture of the proposed two-step g2p conversion system is depicted in Fig. 4.1.

The general-purpose g2p converter creates an initial phonemic transcription which is then corrected by the p2p converter. In order to perform its work, the p2p converter can inspect both the initial phonemic transcription and the orthography of the name it has to process. The heart of the p2p converter is a set of stochastic correction rules, with each rule expressing the following:

[9] This section is largely based on [7].

If a particular phonemic pattern (called the rule input) occurs in the initial phonemic transcription and if the context in which it occurs meets the rule condition, then it may have to be transformed, with a certain firing probability, to an alternative phonemic pattern (called the rule output) in the final transcription.

The rule condition can describe constraints on the identities of the phonemes to the left and the right of the rule input, the stress level of the syllable associated with that input, the position of this syllable in the word, etc. It can also express constraints on the graphemic patterns that gave rise to the rule input and the contextual phonemes.

We distinguish three types of correction rules: (1) stress substitution rules (SS-rules) which replace a stress mark by another (no stress is also considered as a stress mark here), (2) phoneme substitution and deletion rules (PSD-rules) which transform a phonemic pattern into another one (including the empty pattern representing a pattern deletion) and (3) phoneme insertion rules (PI-rules) inserting a phonemic pattern at some position. The linguistic features for describing the context can be different for the respective rule types.

The rewrite rules are implemented in the form of decision trees (DTs). Each DT comprises the rules that apply to a particular rule input. The DTs are learned automatically from training examples by means of machine learning algorithms that were previously applied with success to add pronunciation variants to the lexicon of an automatic speech recognizer.

4.3.2 Learning the Correction Rules

The whole rule learning process is depicted in Fig. 4.2 (cf. also Chap. 14, Sect. 14.2, p. 260).

In general terms, the process is applied to a set of training objects each consisting of an orthography, an initial g2p transcription (called the source transformation), the correct transcription (called the target transcription) and a set of high-level semantic features (e.g. the name type or the language of origin) characterizing the name. Given these training objects, the learning process then proceeds as follows:

1. The objects are supplied to an alignment process incorporating two components: one for lining up the source transcription with the target transcription (sound-to-sound) and one for lining up the source transcription with the orthography (sound-to-letter). These alignments, together with the high-level features are stored in an alignment file.
2. The transformation learner analyzes the alignments and identifies the (focus, output) pairs that are capable of explaining a lot of systematic deviations between the source and the target transcriptions. These pairs define transformations which are stored in a transformation file.
3. The alignment file and the transformation file are supplied to the example generator that locates focus patterns from the transformation file in the source transcriptions, and that generates a file containing the focus, the corresponding

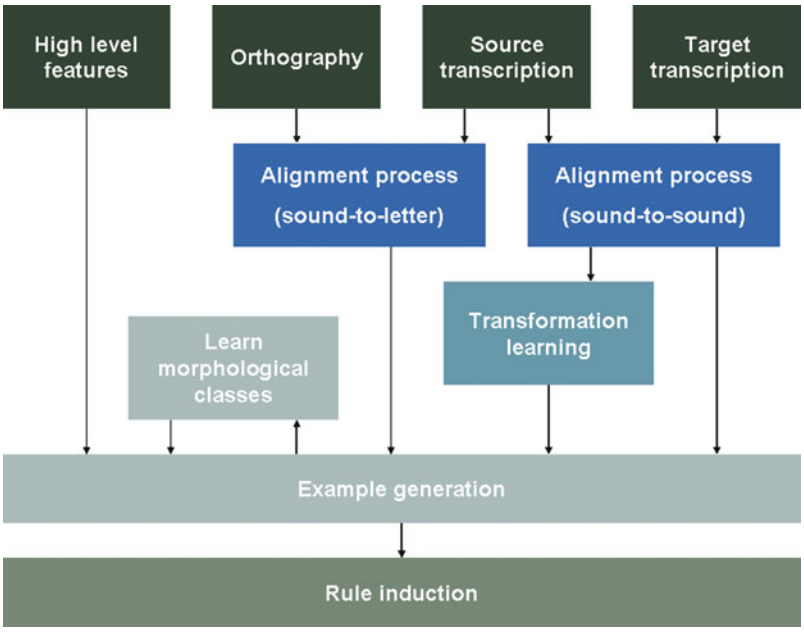

Fig. 4.2 Process for automatically learning of a P2P converter

contextual features and the output for each detected focus pattern. These combinations will serve as the examples from which to train the rules. The example generator also provides statistics about the words whose initial transcription is incorrect, and from these statistics one can create prefix, suffix and syllable sets which define 'morphological' features that can be added to the already mentioned feature set. By running the example generator a second time one creates training examples which also incorporate these 'morphological' features.

4. The example file is finally supplied to the actual rule induction process which automatically constructs a decision tree per focus.

In the subsequent subsections we further elaborate the rule learning process and we also indicate where a manual intervention is possible or desirable. For a more in-depth discussion of the process, the reader is referred to [1] and the documentation provided with the software.

4.3.2.1 Alignment

As indicated before, the alignment process performs a sound-to-letter (or phoneme-to-grapheme) alignment between the source transcription and the orthography, and a sound-to-sound (or phoneme-to-phoneme) alignment between the source and the target phonemic transcription. By replacing every space in the orthography by the

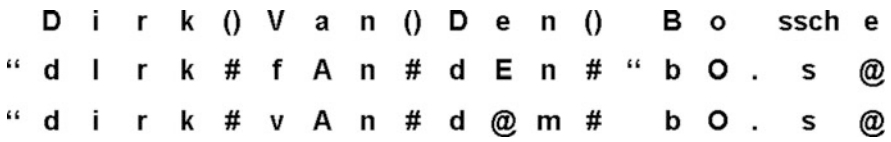

Fig. 4.3 Alignment of the orthography (*top*), the source transcription (*mid*) and the target transcription (*bottom*) of the person name *Dirk Van Den Bossche*

symbol “()”, one can visualize both alignments together in the form of a matrix (cf. Fig. 4.3). The rows subsequently represent the orthography (row 1), the source transcription (row 2) and the target transcription (row 3). The alignment between a source transcription and a destination transcription (either the orthography or the target phonemic transcription) is obtained by means of Dynamic Programming (DP) which is controlled by a predefined image set per unit that can appear in the source transcription and some easy to set control parameters. The image set of a source unit comprises all the units that can appear in a target transcription and that frequently co-occur with the source unit . Since it is generally known that certain graphemic patterns (e.g. “eau”, “ie”, “ij”, etc. in Dutch) often give rise to one sound, the sound-to-letter alignment can align a sound to sequences of up to four graphemes. Figure 4.3 shows a multi-character pattern “ssch” which is lined up with the source phoneme /s/. Since the image sets mostly represent domain independent knowledge, good baseline sets for a certain language can be constructed once, and later be reused for different domains. The user then has the opportunity to update the files manually on the basis of statistical information (most frequently observed sound-to-sound and sound-to-letter substitutions, number of deletions, insertions and substitutions within and outside the image sets) and to repeat the alignments with these new files.

4.3.2.2 Transformation Retrieval

In a second stage, the outputs of the aligner are analyzed in order to identify the (focus,output) transformations that can explain a large part of the observed discrepancies between the source transcriptions and the corresponding target transcriptions. Since stress markers are always lined up with stress markers (cf. previous section), and since every syllable is presumed to have a stress level of 0 (no stress), 1 (secondary stress) or 2 (primary stress), the stress transformations are restricted to stress substitutions. All of the six possible substitutions that occur frequently enough are retained as candidate stress transformations. The candidate phonemic transformations are retrieved from the computed alignments after removal of the stress markers. That retrieval process is governed by the following principles:

1. Consecutive source phonemes that differ from their corresponding target phonemes are kept together to form a single focus,
2. This agglomeration process is not interrupted by the appearance of a matching boundary pair (as we also want to model cross-syllable phenomena),

Fig. 4.4 Candidate transformations that can be retrieved from the alignment of Fig. 4.3

3. A focus may comprise a boundary symbol, but it cannot start/end with such a symbol (as we only attempt to learn boundary displacement rules, no boundary deletion or insertion rules),
4. (Focus,output) pairs are not retained if the lengths of focus and output are too unbalanced (a ratio >3), or if they imply the deletion/insertion of three or more consecutive phonemes,
5. (Focus,output) pairs not passing the unbalance test are split into two shorter candidate transformations whenever possible.

Once all utterances are processed, the set of discovered transformations is pruned on the basis of the phoneme discrepancy counts associated with these transformations. The phoneme discrepancy count expresses how many source phonemes would become equal to their corresponding target phoneme if the transformation were applied at the places where it helps (and not at any other place). Figure 4.4 shows one stress transformation (from primary to no stress) and three phonemic transformations (/I/,/i/), (/f/,/v/) and (/E n/,/@ m/) that comply with the five mentioned principles and that emerge from the alignment of Fig. 4.3.

4.3.2.3 Example Generation

Once the relevant transformation list is available, the focuses appearing in that list are used to segment the source transformation of each training object. The segmentation is performed by means of a stochastic automaton. This automaton represents a unigram model that comprises a set of phoneme consuming branches. Each branch corresponds to a single or multi-state focus model containing states to consume the subsequent phonemic symbols of the focus it represents. One additional branch represents a single-state garbage model that can consume any phonemic unit. Transition probabilities are automatically set so that a one-symbol focus will be preferred over the garbage model and a multi-state focus model will be preferred over a sequence of single state focus models. Once the segmentation of a source transcription is available, a training example will be generated for each focus segment encountered in that transcription. Recalling that we want to learn three types of correction rules: (1) stress substitution rules (SS-rules), (2) phoneme substitution and deletion rules (PSD-rules) and (3) phoneme insertion rules (PI-rules), we will also have to generate three types of examples. Each example consists of a rule input, a rule output and a set of features describing the linguistic context in which the rule input occurs.

4.3.2.4 Rule Induction

From the training examples, the system finally learns a decision tree for each focus appearing in the transformation list. The stochastic transformation rules are attached to each of the leaf nodes of such a tree. The identity rule (do not perform any transformation) is one of the stochastic rules in each leaf node. The collection of all learned decision trees constitutes the actual P2P converter. The decision trees are grown incrementally by selecting at any time the best node split one can make on the basis of a list of yes/no-questions concerning the transformation context and a node splitting evaluation criterion. The node splitting evaluation criterion is entropy loss. Since it has been shown that more robust trees can be learned if asking questions about whether a feature belongs to particular value class are allowed, we have accommodated the facility to specify such value classes for the different feature types that appear in the linguistic description of the training examples.

4.3.3 The Actual p2p Conversion

If the orthography is involved in the description of the correction rules, the p2p converter starts with performing an alignment between the initial phonemic transcription and the orthography.

The next step is to examine each syllable of the initial transcription and to apply the stress mark modification rules if the conditions are met.

The third step consists of a segmentation of the initial phonemic transcription into modifiable patterns and non-modifiable units (by means of the segmentation system that was applied before during rule induction). Once the segmentation is available, the pronunciation variant generator will try PI rules at the start of each non-empty segment and PSD rules at the start of each modifiable segment. If at a certain point one or more rules can be applied, different variants (including the one in which the input pattern is preserved) can be generated at the corresponding point in already created partial variants [7]. The output of the pronunciation variant generator is a tree shaped network representing different phonemic transcriptions with different attached probabilities. The p2p converter will select the transcription with the highest probability as the final phonemic transcription. Obviously, one can expect that in a number of cases this transcription will be identical to the initial transcription.

4.3.4 The Transcription Tool

In their simplest operation mode the AUTONOMATA transcription tools aim at providing phonetic transcriptions for a list of orthographic items, either words or sentences. The transcriptions are either generated by an embedded **standard g2p converter** of Nuance (see below), or by a tandem system also comprising

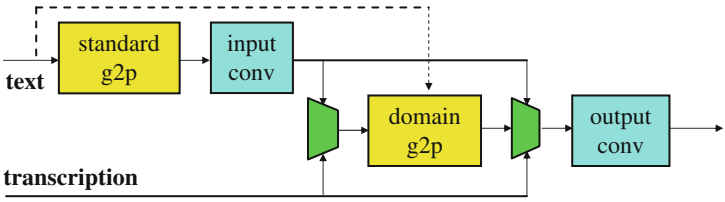

Fig. 4.5 Text-to-phoneme conversion in the autonomata transcription tools

a **domain specific phoneme-to-phoneme (p2p) converter** which tries to correct some of the mistakes made by the standard g2p converter and which also has access to the orthography. A third possibility is to select an already available phonetic transcription from the input file and to supply that to the p2p converter or to the **output conversion** block. In order to allow for a flexible use of different phonemic alphabets like CGN, LH+ (generated by the Nuance g2p converters) and YAPA (used in the HMM75 speech recognition engine), the transcription tools implement the transcription process depicted in Fig. 4.5.

The g2p always produces a transcription in terms of LH+ phonemes. The p2p converter can work directly on the g2p output or on a transformed version of it (e.g. transformed to CGN) obtained by an **input conversion** block. In the first case one does not have to specify any input conversion, in the other case one must specify one as explained below. The transcription tool can either select the transformed g2p transcription, the dedicated p2p output or a selected transcription from the input file. If needed, it can perform an additional conversion of this transcription, for instance, if the output transcriptions must be used in combination with a speech recognizer that is working with yet another phonemic alphabet.

In summary one can discern three phonetic alphabets: the **g2p-alphabet** (always LH+), the **p2p-alphabet** (LH+ or anything else being defined by the input conversion) and the **output-alphabet** (the p2p-alphabet or anything else being defined by the output conversion). In the simplest case all these alphabets are the same (LH+).

Since the p2p converters can operate on the output of the general-purpose g2p converter as well as on any automatic transcription that is already available in the input lexicon, it is easy to implement a cascade of p2p converters (let the first one operate on the g2p-output, the second one on the automatic transcription that was produced by the first p2p converter, etc.)

The transcription tool box is available via the Dutch HLT-agency.[10]

[10]See http://www.tst-centrale.org/nl/producten/tools/autonomata-transcriptietoolset/8-34

4.4 The Autonomata P2P Converters

The transcription tool comes with a number of p2p converters, yielding output in the CGN alphabet, for converting Dutch and Flemish person and place names:

- GeoNames_DUN: to use for Dutch geographical names in combination with DUN version of the Nuance g2p.
- GeoNames_DUB: to use for Flemish geographical names in combination with DUB version of the Nuance g2p.
- PersonNames_DUN: to use for Dutch person names in combination with DUN version of the Nuance g2p.
- PersonNames_DUB: to use for Flemish person names in combination with DUB version of the Nuance g2p.

Furthermore there are p2p converters for Points of Interest (POIs) developed in the Autonomata Too project:

- DUT_POI: to use in combination with DUN version of the Nuance g2p.
- ENG_POI: to use in combination with ENG version of the Nuance g2p.
- FRA_POI: to use in combination with FRF version of the Nuance g2p.

4.5 The Autonomata TOO POI Corpus

The Autonomata POI-corpus[11] was intended as an evaluation corpus for testing p2p converters developed for the transcription of Points of Interest (POIs) such as restaurants, hotels and rental companies. Such names often contain parts with an archaic or otherwise non-standard spelling as well as parts exhibiting a high degree of foreign influence.

4.5.1 Speakers

The corpus contains recordings of native speakers of Dutch, English, French, Turkish and Moroccan. The Dutch group consists of speakers from The Netherlands and Flanders. The English group contains speakers from the United States, Canada, Australia, Great Britain and Hong Kong. The other three groups consist of French, Turkish and Moroccan people, respectively. Table 4.3 contains the number of speakers in each group. Native speakers of Dutch will be referred to as Dutch speakers, speakers of foreign languages as foreign speakers. For both groups, this is a reference to native language, not to nationality.
Gender Speakers are equally distributed over age: 40 male and 40 female.

[11] This section is largely based on the corpus documentation written by Marijn Schraagen.

Table 4.3 Speaker distribution in the Autonomata TOO POI corpus

Mother tongue	Number of speakers
Dutch (Netherlands)	20
Dutch (Flanders)	20
English	10
French	10
Turkish	10
Moroccan	10
Total	80

Age All speakers are adults (above 18 years of age). Two categories are defined: younger than 40 years and 40 years or older. The Dutch speakers of each region (Netherlands and Flanders) are equally spread among these two groups. For foreign speakers, age has not been a strict criterion due to practical reasons.

Dialect region The dialect region is defined as in the ASNC and is only applicable to Dutch speakers

Education Education level is divided in two categories: high and low. The first category contains colleges (university and Dutch HBO schools), the second category contains all other levels of education. This variable has no strict distribution.

Home language The language spoken in informal situations is defined for Dutch speakers only. We distinguish three categories: standard Dutch, dialect, or a combination of standard Dutch and dialect. The assessment of this variable is left to the speaker, no criteria are defined for borders between the categories.

Number of years in Dutch language area and language proficiency For foreign speakers, the number of years they have lived in the Dutch language area is recorded. Besides this, we have asked all foreign speakers whether they have attended a language course for Dutch. If available, we have recorded the CEF level (Common European Framework for language proficiency). If the CEF level was not known by the speaker, we have indicated whether or not a language course was attended.

Foreign language proficiency All speakers were asked what languages they speak, and how proficient they are in every language: basic, intermediate, or fluent. The assessment of level is left to the speakers.

4.5.2 Recording Material

The POI list is designed in order to present 200 POI's to each speaker. The POI's are Hotel-Motel-Camp site and Café-Restaurant-Nightlife names that have been selected from the TeleAtlas POI database of real Points-of-Interest in The Netherlands and Belgium. The POI names were selected according to language. The list contains Dutch, English and French names, and names in a combination of either Dutch and English or Dutch and French.

Table 4.4 Reading lists with number of prompted items per speaker group

Speaker group	Number of speakers	Number of POI names		
		DU $_{50 = 30\,DU + 10\,(DU+EN) + 10\,(DU+FR)}$	EN	FR
Dutch group 1	10	50 (A)	75	75
Dutch group 2	10	50 (B)	75	75
Dutch group 3	10	50 (C)	75	75
Dutch group 4	10	50 (D)	75	75
Foreign	40	50 (A)+50 (B) + 50 (C) + 50 (D)	0	0

All foreign speakers read the same list, containing Dutch names and combination names. The focus of the Autonomata TOO research project did not require to present English or French names to foreign speakers.

Dutch speakers read one of four lists. Every list contains a unique selection of English and French names. Besides this, all Dutch lists contained a quarter of the Dutch names and the combination names from the foreign list. Table 4.4 shows the POI list construction with the number of names in each category. The colors and characters A–D indicate the list the names belong to.

Following this division, every Dutch name (including combination names) is spoken by 50 different speakers (all foreign speakers and 10 out of 40 Dutch speakers). Every French and English name is spoken by ten different speakers. The total list contains 800 names, of which 200 Dutch (120 Dutch only and 80 combination names with English or French), 300 English and 300 French names.

4.5.3 Recording Equipment and Procedure

The recordings were made using a software tool by Nuance, specially developed for the Autonomata TOO project and built as a GUI around the Nuance VoCon 3200 speech recognition engine, version 3.0F3. The speech recognition engine used a Dutch grapheme to phoneme converter from Nuance, and as a lexicon the full TeleAtlas POI set for The Netherlands and Belgium. The recognition engine used a baseline system of Dutch g2p transcriptions and Dutch monolingual acoustic models.

The recordings were made on a laptop with a USB headset microphone. Digital recordings were stored on the laptop hard disk in 16 bit linear PCM (wav-format). The sampling frequency is 16 kHz. We used a unidirectional Electret condensor microphone with a frequency range of 40–16 kHz.

The speaker was in control of the application. A POI name was shown on the screen, and the speaker started recording this name by pressing a button. Speech recognition was performed immediately on starting the recording, and the recognition result was shown to the speaker. The system checked whether the POI name was recognized correctly. On successful recognition, the system proceeded

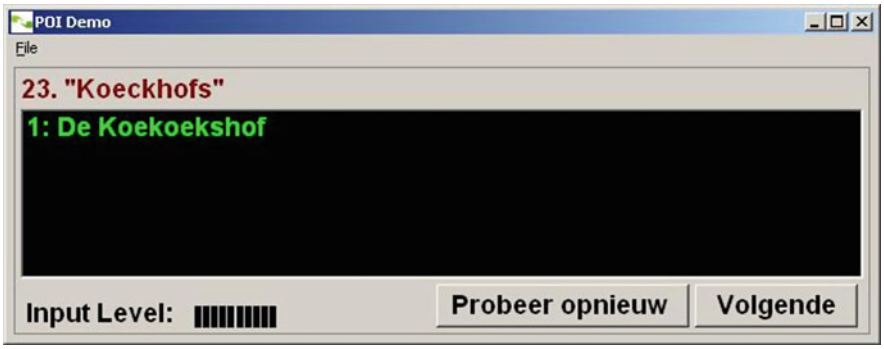

Fig. 4.6 Screenshot of the Autonomata POI database recording tool

to the next POI name. On failed recognition, the speaker was presented with a possibility to do an additional recording. The recognition result for the new attempt was again presented to the user. This process repeated itself until either the POI name was either recognized correctly or the user decided to proceed to the next item. The speaker could choose how many times s/he wanted to repeat a failed utterance, with a minimum of one repetition. This was to create a situation which is similar to using a real application in which a speaker will re-try and adopt after a misrecognized utterance [3]. All utterances were kept and stored on hard disk.

The screenshot in Fig. 4.6 illustrates the recording tool. An example is shown where the recognition failed.

The researcher was sitting next to the test subject during the entire experiment, to assist in using the recording tool and to control the process of repeating utterances. Any instruction during the experiment was aimed to improve the quality of the recording (such as preventing incomplete recordings or deliberately incorrect pronunciations), and to prevent useless repetitions (for example repeating an utterance more than three times in exactly the same way). The researcher did not answer questions regarding the correct pronunciation of an item. Before starting the real recording sessions, a number of ten test recordings was performed to let the user get acquainted to the recording tool works and to monitor the quality of the recordings. After the recording session, all recordings were checked. Incomplete recordings or recordings containing a severely mixed up reading were deleted.

The recordings were performed in sound-proof studios in Utrecht and Ghent. If test subjects were unable or unwilling to come to the record studio, the recording was performed on location. In this case, we have tried to minimize any influence from outside noise and reverberation.

4.5.4 Annotations

Each name token comes with an orthographical representation and an auditorily verified phonemic transcription (containing LH+ phonemes, word and syllable boundaries and primary stress markers). The latter were made in Praat in very much the same way as in the ASNC. The transcription protocol that was used is distributed together with the corpus.

4.5.5 Corpus Distribution

The corpus is 1.7 GB large and is distributed by the Dutch HLT-agency (TST-centrale).[12] The corpus documentation is very much similar to the one of the ASNC, but the POI-corpus also contains the ASR recognition results obtained with the Nuance VoCon 3200 recognition engine (version 3.0F1) during the recording process.

References

1. Réveil, B., Martens, J.-P., Van den Heuvel, H.: Improving proper name recognition by means of automatically learned pronunciation variants. Speech Commun. **54**(3), 321–340 (2012)
2. Schraagen, M., Bloothooft, G.: Evaluating repetitions, or how to improve your multilingual ASR system by doing nothing. Proceedings LREC2010, Malta (2010)
3. Van den Heuvel, H., Martens, J.-P., Konings, N.: G2P conversion of names. What can we do (better)? Proceedings Interspeech, Antwerp, pp. 1773–1776 (2007)
4. Van den Heuvel, H., Martens, J.-P., D'hoore, B., D'hanens, K., Konings, N.: The Autonomata Spoken Name Corpus. Design, recording, transcription and distribution of the corpus. Proceedings LREC 2008, Marrakech (2008)
5. Van den Heuvel, H., Martens, J.-P., Konings, N.: Fast and easy development of pronunciation lexicons for names. In: Proceedings LangTech 2008, Rome (2008)
6. van den Heuvel, H., Réveil, B., Martens, J.-P., D'hoore, B.: Pronunciation-based ASR for names. In: Proceedings Interspeech2009, Brighton (2009)
7. Yang, Q., Martens, J.-P., Konings, N., Van den Heuvel, H.: Development of a phoneme-to-phoneme (p2p) converter to improve the grapheme-to-phoneme (g2p) conversion of names. In: Proceedings LREC, Genua, pp. 287–292 (2006)

[12] See: http://www.tst-centrale.org/nl/producten/corpora/autonomata-poi-corpus/6-70